

Customer No. 000959

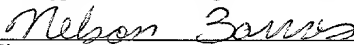
Case Docket No. SMQ-011

THE ASSISTANT COMMISSIONER FOR PATENTS
Box Patent Application
Washington, D.C. 20231

"Express Mail" Mailing Label Number EL 008 489 134 US

Date of Deposit March 16, 1999

I hereby certify that this transmittal letter and the papers referred to as being enclosed therein are being deposited with the United States Postal Service "Express Mail Post Office to Addressee" service under 37 CFR 1.10 on the date indicated above and is addressed to the Assistant Commissioner for Patents, Box Patent Application, Washington, DC 20231.



Signature

Nelson F. Barros

Please Print Name of Person Signing

JC542 U.S. PTO

09/270141

03/16/99

Sir:

Transmitted herewith for filing is the patent application of

Inventor(s): Gary D. Foster

For: LIGHTWEIGHT GLOBAL DISTRIBUTION MECHANISM

Enclosed are:

- ☐ This is a request for filing a ☐ continuation ☐ divisional application under 37 CFR 1.53(b), of pending prior application serial no. _____ filed on _____ entitled _____.
- ☒ 9 pages of specification, 7 pages of claims, 1 pages of abstract.
- ☒ 12 sheets of drawings (Figures 1-12).
- ☒ An Executed Declaration, Petition and Power of Attorney.
- ☒ An assignment of the invention to Sun Microsystems, Inc. A recordation form cover sheet (Form PTO 1595) is also enclosed.
- ☐ A verified statement to establish small entity status under 37 C.F.R. 1.9 and 37 C.F.R. 1.27.
- ☐ Other _____

The filing fee has been calculated as shown below:

	(Col. 1)	(Col. 2)
FOR:	NO. FILED	NO. EXTRA
BASIC FEE	////////////////////	
TOTAL CLAIMS	52 - 20	= 32
INDEP. CLAIMS	13 - 3	= 10
<input type="checkbox"/> MULTIPLE DEPENDENT CLAIMS PRESENTED		

* If the difference in Col. 2 is less than zero,
enter "0" in Col. 2.

SMALL ENTITY	
RATE	FEE
////////	\$
x 9=	\$
x 39	\$
+130	\$
TOTAL	0

OTHER THAN SMALL ENTITY	
RATE	FEE
////////	\$ 760
x 18=	\$ 576
x 78	\$ 780
+260-	\$
TOTAL	\$2116.00

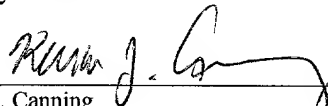
- ☐ Please charge my Deposit Account No. 12-0080 in the amount of \$.
A duplicate copy of this sheet is enclosed.
- ☒ A check in the amount of \$ \$2116.00 to cover the filing fee is enclosed.
- ☒ The Commissioner is hereby authorized to charge payment of the following fees associated with this communication or credit any overpayment to Deposit Account No. 12-0080.
A duplicate copy of this sheet is enclosed.
- ☒ Any additional filing fees required under 37 C.F.R. 1.16.
- ☒ Any patent application processing fees under 37 C.F.R. 1.17.

- ☐ The Commissioner is hereby authorized to charge payment of the following fees during the pendency of this application or credit any overpayment to Deposit Account No. 12-0080.
A duplicate copy of this sheet is enclosed.
- ☐ Any patent application processing fees under 37 C.F.R. 1.17.
- ☐ The issue fee set in 37 C.F.R. 1.18 at or before mailing of the Notice of Allowance, pursuant to 37 C.F.R. 1.311(b).
- ☐ Any filing fees under 37 C.F.R. 1.16 for presentation of extra claims.
- ☒ A check in the amount of \$ 40.00 to cover the recording of assignment documents is also enclosed.
- ☒ Address all future communications (May only be completed by applicant, or attorney or agent of record) to Kevin J. Canning, Esq. at **Customer Number: 000959** whose address is:

Lahive & Cockfield, LLP
28 State Street
Boston, Massachusetts 02109

Date: March 16, 1999

LAHIVE & COCKFIELD, LLP
Attorneys at Law

By 
Kevin J. Canning
Reg. No. 35,470
28 State Street
Boston, MA 02109
(617) 227-7400
Telecopier (617) 742-4214

LIGHTWEIGHT GLOBAL DISTRIBUTION MECHANISM

Technical Field

The present invention relates generally to data processing systems and more particularly to a lightweight global distribution mechanism.

Background of the Invention

In a distributed system there is often a need to distribute packages to various computer systems within the distributed system. These packages may contain programs, patches, documents or other files. In conventional systems, distribution requires the computer systems to register with a master to identify what packages are to be distributed to the respective computer systems. The master then issues a publishing event indicating that a package is available, and the package is distributed to the parties that registered for the package.

Unfortunately, this conventional approach suffers from some drawbacks. For example, there is a great deal of overhead incurred in providing support for the registration. A registry must be maintained, and resources must be provided to enable the computer systems to register with the master. In addition, there is no mechanism for a computer system to anonymously install packages as needed.

Summary of the Invention

The present invention addresses the limitations of conventional systems by providing a lightweight global distribution mechanism for distributing packages. The present invention provides a distribution mechanism that does not rely upon registration, and thus, does not incur the overhead associated with maintaining a registry. In addition, the present invention allows clients to anonymously download or install the packages. The present invention is platform independent such that it may be practiced on a number of different types of platforms.

In accordance with one aspect of the present invention, a method is practiced in a distributed system that includes a publishing master and a client system. The publishing master is provided with an index of available packages for loading. The packages may take many forms, including the form of a patch, a document or a hypertext markup language (HTML) file, for example. The index is accessed on behalf of the client system to identify at least one selected package for installation at the client system. The selected package is then loaded on the client system. The selected package may be loaded without registration of the client system with the publishing master, and the publishing master may be unaware of the loading of the selected package at the client system.

In accordance with a further aspect of the present invention information that identifies packages that may be installed at a client is stored at a server. Data that identifies packages to which the client subscribes is stored at the client. The information identifying the packages that may be installed at the client and the data that identifies packages to which the client subscribes are used to determine which of the packages are to be installed at the client.

In accordance with a further aspect of the present invention, a method is practiced in a computer network that has a server and a client. In accordance with this method, data is provided at the server regarding what packages are available. An itemization of packages to which the client subscribes is provided at the client. The following steps are then repeated multiple times: data is accessed regarding what packages are available, and data is accessed regarding the itemization of packages to which the client subscribes to determine what selected packages to load and subsequently the selected packages are loaded at the client. The steps that are repeated multiple times may be repeated at periodic intervals or upon demand by a user.

In accordance with an additional aspect of the present invention, a distributed system comprises a publishing master that holds an index of packages that are available for installation and a repository for storing the packages that are available for installation. The distributed system also includes a client computer system for identifying selected packages among the packages indexed by the index of the packages. The client computer system obtains the selected packages from the repository and installs the selected packages.

In accordance with another aspect of the present invention, a method is practiced in the distributed system having repositories for storing packages. A selected package is provided on multiple repositories. Response times are determined from the repositories in which the selected package is provided. The selected package is then retrieved from the repository that has the shortest response time. As test communication may be used to assist in determining response times.

In accordance with yet another aspect of the present invention, a method is practiced by a client system within a distributed system. The distributed system includes a publishing master that has an index of packages available for installation. At least a portion of the index is requested from the publishing master. This portion of the index is received at the client system and processed to identify a selected package to install. Subsequently, a copy of the selected package is obtained and the selected package may then be installed.

In accordance with a further aspect of the present invention, a method is practiced in a distributed system that has a client and a storage holding packages that are available for downloading. Data is obtained that identifies packages that are available

for downloading. The packages to which the client subscribes are identified as are the packages that are already downloaded on the client. Selected packages are then downloaded to the client from the storage. The selected packages are those which are identified for installation, identified as packages to which the client subscribed and
5 identified as not yet being downloaded to the client.

Brief Description of the Drawings

An illustrative embodiment consistent with the principles of the present invention will be described below relative to the following drawings.

10 FIGURE 1 depicts a block diagram of a distributed system that is suitable for practicing the illustrative embodiment.

FIGURE 2 illustrates an exemplary client computer system in more detail.

FIGURE 3 is a flow chart illustrating the steps that are performed to add a package to the packages that are available for installation.

15 FIGURE 4 depicts the format of a package index.

FIGURE 5 is a flow chart illustrating the cycle of steps that is performed to keep client computer systems updated.

FIGURE 6 is a flow chart illustrating the steps that are performed to update a single client computer system.

20 FIGURE 7 is a flow chart illustrating the steps that are performed to identify what packages need to be updated on a given client computer system.

FIGURE 8 illustrates how a list of packages that need to be updated is generated from other data.

FIGURE 9 depicts an example of a portion of a subscription list.

25 FIGURE 10 illustrates the steps that are performed to update a given client computer system.

FIGURE 11 depicts the logical organization of an exemplary notification.

FIGURE 12 is a flow chart illustrating the steps performed to dynamically determine a repository to use in retrieving a package for installation when the package is
30 stored on multiple repositories.

Detailed Description of the Invention

The illustrative embodiment provides a lightweight global distribution mechanism. The distribution mechanism is global in that all of the client computer
35 systems within a distributed system may be serviced using this mechanism. The mechanism is lightweight in that it requires minimal overhead. No registration is used and no registry is maintained. The distribution mechanism is anonymous in that a client computer system may install packages unbeknownst to a publishing master.

Figure 1 depicts an example of a distributed system 10 that is suitable for practicing the illustrative embodiment. Those skilled in the art will appreciate that the depiction in Figure 1 is intended to be merely illustrative and not limiting of the present invention. The present invention may be practiced in environments that use a different
5 number of repositories, client systems and publishing masters than shown in Figure 1. In addition, different interconnection arrangements may be provided while still practicing the present invention.

The distributed system 10 includes a publishing master 12. The publishing master 12 is responsible for maintaining a package index that serves as an index of
10 packages that are available for installation. This index itemizes the packages that may be installed at the client systems 14. The publishing master 12 may be a dedicated server computer system. Alternatively, the publishing master may be realized as a process that runs on a computer system along with other processes. Those skilled in the art will appreciate that the publishing master may be implemented using a variety of
15 different types of computer systems including personal computers, workstations, mini-computers or mainframe systems.

The client systems 14 are computer systems on which the packages may be installed or downloaded. A "package," as used herein, refers to a logical quantity of information that may be encapsulated and sent to a client system 14 for installation. A
20 package may contain one or more computer programs, patches, documents, data, records, HTML files or the like. In general, a package may contain any information that is useful for distribution to the client systems 14 within the distributed system 10. As used herein, a "patch" is a section of computer program instructions that is provided to update or repair a deficiency in an existing routine or program. The patches may
25 provide additional features or functions to an existing version of a program between releases of versions of the program. The client systems 14 may take many forms. For example, the client systems 14 may be realized as personal computer systems, as server computer systems, as workstations, as network computers, as intelligent digital devices or as other types of devices. The client systems 14 are the entities on which the
30 packages are installed. Thus, if a package contains a patch, the client system 14 receives the patch and installs the patch therein.

The distributed system 10 may include one or more repositories 16. The repositories 16 may be realized as generic storage servers. The packages that are indexed by the package index are stored on the repositories 16. A single package may
35 be stored on multiple repositories 16. As will be explained in more detail below, the client systems 14 may obtain packages from the repositories 16 and download and/or install them.

The distributed system 10 may be a web-based system, such as the Internet, an intranet or an extranet. In such a case, the publishing master 12 may be realized as a web server that provides the client systems 14 access to the publishing index. The repositories 16 support anonymous File Transfer Protocol (FTP) and/or Hypertext

5 Transfer Protocol (HTTP) access to the packages stored therein.

Figure 2 depicts examples of a format for a client system 14. The client system 14 includes a central processing unit (CPU) for overseeing operations within the client system 14. The CPU 20 executes computer program instructions to perform activities. The client system 14 may include input devices, such as a keyboard 22 and a mouse 24. The client system 14 may also include output devices, such as a video display 26. The client system 14 may include a network adapter 28 for interfacing the client system with a computer network 30. The client system 14 may contain a modem 32 for communicating with remote computing resources. The modem 32 may be realized as a conventional modem, a wireless modem or a cable modem, for example.

15 The client system includes both primary storage 40 and secondary storage 34. The storages 34 and 40 may include removable computer readable medium, including removable magnetic disks, optical disks and/or magneto-optical disks. The secondary storage 34 may hold a number of lists 36 (which will be described in more detail below) that are used in determining which packages to download and/or install. Secondary storage 34 may also hold a copy of a subscription list 38 that identifies the packages to which the client system subscribes (i.e. those packages for which the client system wants updates).

The primary storage 40 may hold a number of different types of programs and data. The primary storage may include a web browser 41 for enabling the client system 14 to communicate with web servers and, in general, to provide web-based access to remote computing resources. The primary storage 40 holds three modules 44, 46 and 48 that perform respective roles in updating packages on the client system 14. As will be described in more detail below, the check-pkgs module 44 is responsible for identifying what packages need to be updated and installed at the client system. The sync-pkgs module 46 is responsible for performing the updates, and the notification module 48 is responsible for generating notifications that notify the completion of updates.

The client system 14 may run an operating system, such as the Solaris™ operating system from Sun Microsystems, Inc. of Palo Alto, California. The operating system may include a scheduler 50 (e.g. Cron) that is responsible for scheduling activities within the client system 14. As will be described in more detail below, the scheduler may be responsible for triggering updates so that the packages that are installed on the client system 14 may be ensured to be current. The client system may

include various transmission media for transmitting signals to perform desired operations.

Those skilled in the art will appreciate that the depiction of a client system 14 shown in Figure 2 is intended to be merely illustrative and not limiting of the present invention. Each of the client systems 14 need not have the same configuration. Moreover, the present invention may be practiced with computer systems that include different components than those depicted in Figure 2.

The global distribution mechanism of the illustrative embodiment facilitates the addition of packages so that the packages are available for installation. Figure 3 depicts a flow chart of the steps that are performed to add a package. Initially, the package is staged on a repository 16 (step 52 in Figure 3). The choice of repository for storing the package may be based upon a number of factors, including available storage capacity, proximity to subscriber client systems 14 and load balancing considerations. For example, a single package may be stored on multiple repositories 16, and a client system may dynamically choose what package to download based on current loads, as will be described in more detail below. The package index stored on the publishing master 12 is then updated to include a listing for the newly staged package (step 54 in Figure 3). Conversely, the distribution mechanism facilitates the removal of packages by removing the packages from the index, and may entail deletion of the package from the repositories 16.

Figure 4 depicts the logical format of the package index 60. The package index includes a number of entries where each entry is associated with a particular package. In Figure 4, entries 70, 72, 74 and 75 are associated with separate packages. Each entry includes a name field 62 that identifies the name of the package (e.g. coyote). Each entry also includes a version field 64 that identifies the version of the package (e.g. 2.0). Each entry additionally includes a type field 66 that identifies the type of package (e.g. "O") and a location field 68 that identifies a location of the package (e.g. <http://www.arizona.com/coyote>). Multiple locations may be specified (see entry 75) where the package is stored at multiple locations.

The type field 66 may specify whether the package is a required update ("R"), an optional update ("O"), a patch update ("P") or a stealth update ("S"). A required update is a package that is flagged for update on any system that either does not currently have the package installed or that has a version lower than the version indicated in the package index field. An optional update is flagged for update on any system that has subscribed to the package and that does not have the package installed or has a version lower than the version indicated in the package index field. A patch update can either be a required package or an optional package. Required patch packages are by default installed on every system. Optional patch packages are only installed on client systems

that subscribe to the patch. The "P" designator in the type field 66 may be followed by a hyphen and an optional package name. In such a case, the patch 11 be updated only if the package with the specified optional package name is currently installed and subscribed to by the client system. A stealth update is a required update to an optional package.

Figure 5 is a flow chart illustrating in the steps that are performed to update packages on a given client system. Initially, the update is triggered (step 80 in Figure 5). The update may be triggered at periodic intervals by the scheduler 50 or may be triggered in a different fashion. For example, the update may be triggered by a user explicitly requesting an update or by other events that indicate that update is required. The update is then performed to install packages as needed (step 82 in Figure 5). In particular, the modules 44, 46 and 48 are executed. The system then waits (step 84 in Figure 5) until an update is again triggered (see step 80 in Figure 5).

Figure 6 is a flow chart illustrating the steps that are performed to update a client system 14. Initially the check-pkgs module 44 is activated to determine what packages need to be updated (step 90 in Figure 6). Figure 7 is a flow chart that illustrates the steps that are performed by the check-pkgs module. Initially, a client system 14 retrieves the package index 60 from the publishing master 12 and generates a list of the published state of the packages (step 100 in Figure 7). Thus, as shown in Figure 8, the package index 110 is retrieved by the client system 14 and processed to generate a publication list 116. The resulting process list reflects the published state of all packages (i.e. a list of packages and versions). The client system then initiates a query to generate a list that identifies the state of all packages installed on the client system 14, including version information for these packages. As shown in Figure 8, the client system 112 generates an installed list 118 that identifies the state of the packages installed on the system.

The client system examines the local subscription list 114 (Figure 8) to generate a list of subscribed packages 120 (step 104 in Figure 7). Figure 8 depicts the format of the local subscription list 114. The local subscription list 114 is a serial listing of the packages to which the local system subscribes.

Those skilled in the art will appreciate that the subscription state list 120 may have a number of different formats and need not be organized as a list per se but may be organized as a set of records, a table or the like. Moreover, the lists may be combined into a single list or in different combinations of lists.

The lists 116, 118 and 120 are then processed to generate a final list 122 that identifies the packages that need to be installed to complete the update (step 106 in Figure 7). Any required updates which are either not installed or have published versions higher than the currently installed packages will be included in the final list 122. Any optional packages which are either not installed or have published versions

higher than the currently installed versions and exist in the local subscription list will also be included on the final list 122.

Once the check-pkgs module has completed generation of the final list 122 (see step 90 in Figure 6). The sync-pkgs module is activated (step 92 in Figure 6). Figure 10 provides a flow chart of the steps performed by this module 46. The module 46 obtains the next package name from the final list (step 120 in Figure 10) and installs the name package from the repository on which the package is located. As was mentioned above, the package index contains location information. The package may be installed using FTP or HTTP (step 122 in Figure 10). This process continues in this fashion until the list is fully processed (see step 124 in Figure 10). Thus, the packages are located on the repositories, downloaded to the client systems and installed.

Notifications are generated (step 94 in Figure 6) to notify appropriate personnel via a desired channel (e.g. via email) that an update has happened. Figure 11 depicts a format of a suitable format for a notification 130. A notification should identify the package name 132 that was installed at the client system 14. The notification 130 should also identify the repository server 144 from which the update was obtained and the notification 130 should include a status field 136 that indicates whether there were any errors or not.

Each of the client systems 14 may perform the update process separately. Alternatively, the client systems 14 may be triggered to perform updates by a master mechanism such that all client systems are updated at the same time.

The illustrative embodiment also facilitates dynamic load balancing. In particular, the illustrative embodiment may store a package on multiple repositories and download a copy of the package from one of the repositories based upon current load conditions. Figure 12 is a flow chart illustrating the steps that are performed to realize such dynamic load balancing in the illustrative embodiment. Initially, the client system 14 accesses the package index 60 and determines that there are multiple location fields 68 (step 140 in Figure 12). These locations are on separate repositories 16. The client system then takes steps to determine the respective loads on the repositories where the package is stored. Specifically, the client sends out a test package to each identified repository (step 142 in Figure 12). The test package received by the respective repositories and returned. The responses are received at the client system for the repositories and based upon the time that has elapsed between when the test packet was sent out to the respective repositories and when the respective responses were received, the client system 14 determines response times for the respective repositories (step 144 in Figure 12). The system retrieves the package from the repository with the shortest response time (step 146 in Figure 12). It is presumed that the repository with the shortest response time is the least loaded and best able to fulfill the request.

The dynamic load balancing provided in the illustrative embodiment allows the strategic location repositories across the network. There is no need for the repository to be crafted for one particular client. The client chooses the repository that is best suited for the client needs. In addition, the scheme allows a server to go down or be services
5 without fear of endangering the client systems 14.

Those skilled in the art will appreciate that the present invention may be practiced with implementations that use a different programmatic structure. The respective phases of the updates need not be performed by separate modules but rather may be performed by a single integrated program or by multiple disparate programs.

10 While the present invention has been described with reference to an illustrative embodiment thereof, those skilled in the art will appreciate that various changes in form and detail may be made without departing from the intended scope of the present invention as defined in the appended claims.

11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

Claims

1. In a distributed system having a publishing master and a client system, a method comprising:
 - 5 providing at the publishing master an index of available packages for loading;
 - accessing the index on behalf of the client system to identify at least one selected package for loading at the client system; and
 - loading the selected package at the client system.
- 10 2. The method of claim 1 wherein the selected package is loaded without registration of the client system with the publishing master.
3. The method of claim 1 wherein the publishing master is unaware of the loading
- 15 of the selected package at the client system.
4. The method of claim 1 wherein the selected package includes a patch for a computer program.
- 20 5. The method of claim 1 further comprising the step of storing the selected package at a repository that is separate from the publishing master so that clients may access and load the selected package.
6. The method of claim 1 wherein the selected package includes a document.
- 25 7. The method of claim 1 wherein the selected package includes at least one hypertext markup language (HTML) file.
8. The method claim 1 wherein the publishing master is a web server.
- 30 9. In a distributed system having a server and a client, a method comprising:
 - storing at the server information identifying packages that may be installed at the client;
 - storing at the client data that identifies packages to which the client
 - 35 subscribes; and
 - using the information identifying packages that may be installed at the client and the data that identifies packages to which the client subscribes to determine which of the packages are to be installed at the client.

10. The method of claim 9 further comprising installing the determined packages at the client.

5 11. The method of claim 9 further comprising providing data regarding what packages have already been installed at the client and using the data regarding what packages have already been installed in determining which of the packages are to be installed at the client.

10 12. The method of claim 9 wherein the information identifying packages that may be installed at the client includes information identifying names and version of the packages.

15 13. The method of claim 9 wherein the determining of which the packages to install at the client comprises generating a list of packages that may be installed at the client to which the client wants to subscribe and which have not already been installed at the client.

20 14. In a computer network having a server and a client, a method comprising the computer-implemented steps of:

(a) providing at the server data regarding what packages are available for installation;

(b) providing at the client an itemization of packages to which the client subscribes; and

25 (c) with the client, repeating multiple times the following steps:

(i) accessing the data regarding what packages are available and accessing the itemization of packages to which the client subscribes to determine what selected packages to install; and

(ii) installing selected packages at the client.

30 15. The method of claim 14 wherein steps (c)(i) and (c)(ii) are repeated at periodic intervals.

35 16. The method of claim 14 wherein steps (c)(i) and (c)(ii) are repeated upon demand by a user.

17. In a distributed system having a publishing master, at least one repository for holding packages and multiple client computer systems, a method comprising:

providing a package index at the publishing master, wherein the package index identifies names and versions of packages stored on the repository;

providing a subscription list at each client system, wherein each subscription list identifies the packages to which the client system subscribes;
5 and

for each client system, at periodic intervals:

(i) identifying selected packages to be loaded by referencing the package index and the subscription list; and

(ii) loading the selected packages at the client system

10 18. The method of claim 17 wherein at least one of the packages that is loaded includes at least one patch.

15 19. The method of claim 18 wherein at least one of the packages that is loaded includes multiple patches.

20. A distributed system, comprising:

a publishing master holding an index of packages that are available for installation;

20 a repository for storing the packages that are available for installation;
and

a client computer system for identifying selected packages among the packages indexed by the index of packages, obtaining the selected packages from the repository and installing the selected packages at the client computer system.

25 21. The distributed computer system of claim 20 wherein the client computer system maintains a subscription list of packages to which the client computer system subscribes.

30 22. The distributed computer system of claim 20 further comprising an additional repository for storing at least a portion of the packages that are available for installation.

23. The distributed computer system of claim 20 further comprising an additional client computer system for identifying a subset of the packages indexed by the index of packages, obtaining the subset of packages from repository and installing the subset of
35 packages at the additional client computer system.

24. In a distributed system having a publishing master, packages available for installation and an index of the packages available for installation which is stored at the publishing master, a client computer system, comprising:

- 5 means for accessing the index on behalf of the client system to identify at least one selected package for installation at the client system; and
means for installing the selected package at the client system.

25. The client computer system of claim 24 wherein the selected package is installed without registration for the client system with the publishing master.

10

26. The client computer system of claim 24 wherein the publishing master is unaware of the installing of the selected package at the client system.

27. The client computer system of claim 24 wherein the selected package includes a patch for a computer program.

15

28. The client computer system of claim 24 wherein the selected package includes a document.

29. The client computer system of claim 24 wherein the selected package includes at least one hypertext markup language (HTML) file.

20

30. In a distributed system having repositories for storing packages, a method comprising:

25

- providing a selected package or multiple repositories;
determining respective response times for the repositories on which the selected package is provided; and
retrieving the selected package from the repository that has the shortest response time.

30

31. The method of claim 30 wherein the determining of respective response times comprises sending a test communication to the repositories on which the selected package is provided and measuring response times to the test communication.

32. The method of claim 30 wherein a client system that wishes to retrieve the selected package performs the determining and the retrieving.

35

33. The method of claim 30 wherein the selected package contains computer instructions.

34. The method of claim 30 wherein the determining of response times is done when
5 the selected package is to be retrieved.

35. In a distributed system having a publishing master holding a package index of packages, and a client system holding a subscription list that identifies packages to which the client system subscribes, a computer-readable medium holding computer-executable instructions for performing a method, comprising:

identifying selected packages to be loaded at the client system by
referencing the package index and the subscription list; and
loading the selected packages at the client system.

36. The computer-readable medium of claim 35 wherein at least one of the packages loaded at the client system includes at least one patch.

37. In a distributed system having a publishing master holding a package index of packages, and a client system holding a subscription list that identifies packages to which the client system subscribes, a transmission medium for transmitting signals for performing a method, comprising:

identifying selected packages to be loaded at the client system by
referencing the package index and the subscription list; and
loading the selected packages at the client system.

38. In a distributed system having a publishing master with an index of packages available for installation and a client system, a method performed by the client system comprising:

requesting at least a portion of the index from the publishing master;
30 receiving at least a portion of the index from the publishing master in response to the request;
processing the received portion of the index to identify a selected package to install; and
obtaining a copy of the selected package.

39. The method of claim 38 further comprising the step of installing the package at the client system.

40. The method of claim 38 wherein the selected package is obtained without registration of the client system with the publishing master.

5 41. The method of claim 38 wherein the selected package contains a patch.

42. The method of claim 38 wherein the selected package contains a document.

43. In a distributed system having a publishing master having an index of packages
10 available for installation and a client system, a computer-readable medium holding computer-executable instructions for performing a method performed by the client system comprising:

requesting at least a portion of the index from the publishing master;
receiving at least a portion of the index from the publishing master in
15 response to the request;
processing the received portion of the index to identify a selected package to install; and
obtaining a copy of the selected package.

20 44. The storage medium of claim 43 wherein the selected package is obtained without registration of the client system with the publishing master.

45. The storage medium of claim 43 wherein the selected package contains a patch.

25 46. The storage medium of claim 41 wherein the selected package contains a document.

47. In a distributed system having storage holding packages that are available for
downloading and a client, a method performed by the client, comprising:
30 obtaining data that identifies packages that are available for downloading;
identifying packages to which the client subscribes;
identifying packages that are already downloaded to the client;
downloading from the storage selected packages at the client, wherein the
selected packages are those which are identified for installation, identified as
35 packages to which the client subscribes and identified as not yet being
downloaded to the client.

48. The method of claim 47 wherein at least one of the packages that is downloaded contains a patch.

49. The method of claim 47 wherein at least one of the packages that is downloaded
5 contains a hypertext document.

50. In a distributed system having storage holding packages that are available for downloading and a client, a computer-readable medium containing computer-executable instructions for performing a method performed by the client, comprising:

10 obtaining data that identifies packages that are available for downloading;
identifying packages to which the client subscribes;
identifying packages that are already downloaded to the client;
downloading from the storage selected packages at the client, wherein the
selected packages are those which are identified for installation, identified as
15 packages to which the client subscribes and identified as not yet being
downloaded to the client.

51. The storage medium of claim 50 wherein at least one of the packages that is
downloaded contains a patch.

52. The storage medium of claim 50 wherein at least one of the packages that is
downloaded contains a hypertext document.

Abstract

- 5 A lightweight global distribution mechanism distributes packages within a distributed system. The packages may hold any type of content and may hold contents such as patches, programs, documents or files. The distribution mechanism is lightweight in that it incurs minimal computational and memory overhead. With the distribution mechanism, a package index that identifies packages for installation is
- 10 stored at a publishing master. Client systems may access and analyze the package index to determine which packages to install. The client systems may hold subscription lists that identify the packages to which the clients subscribe. The packages need not be stored at the publishing master but rather may be stored at separate repositories.

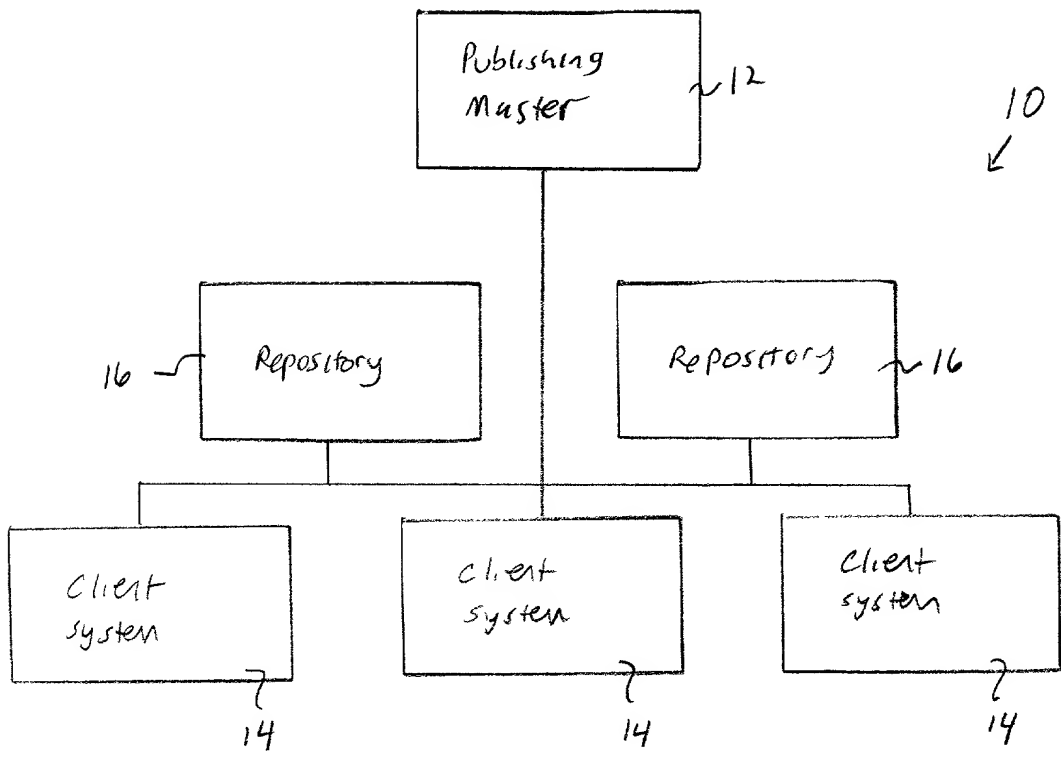
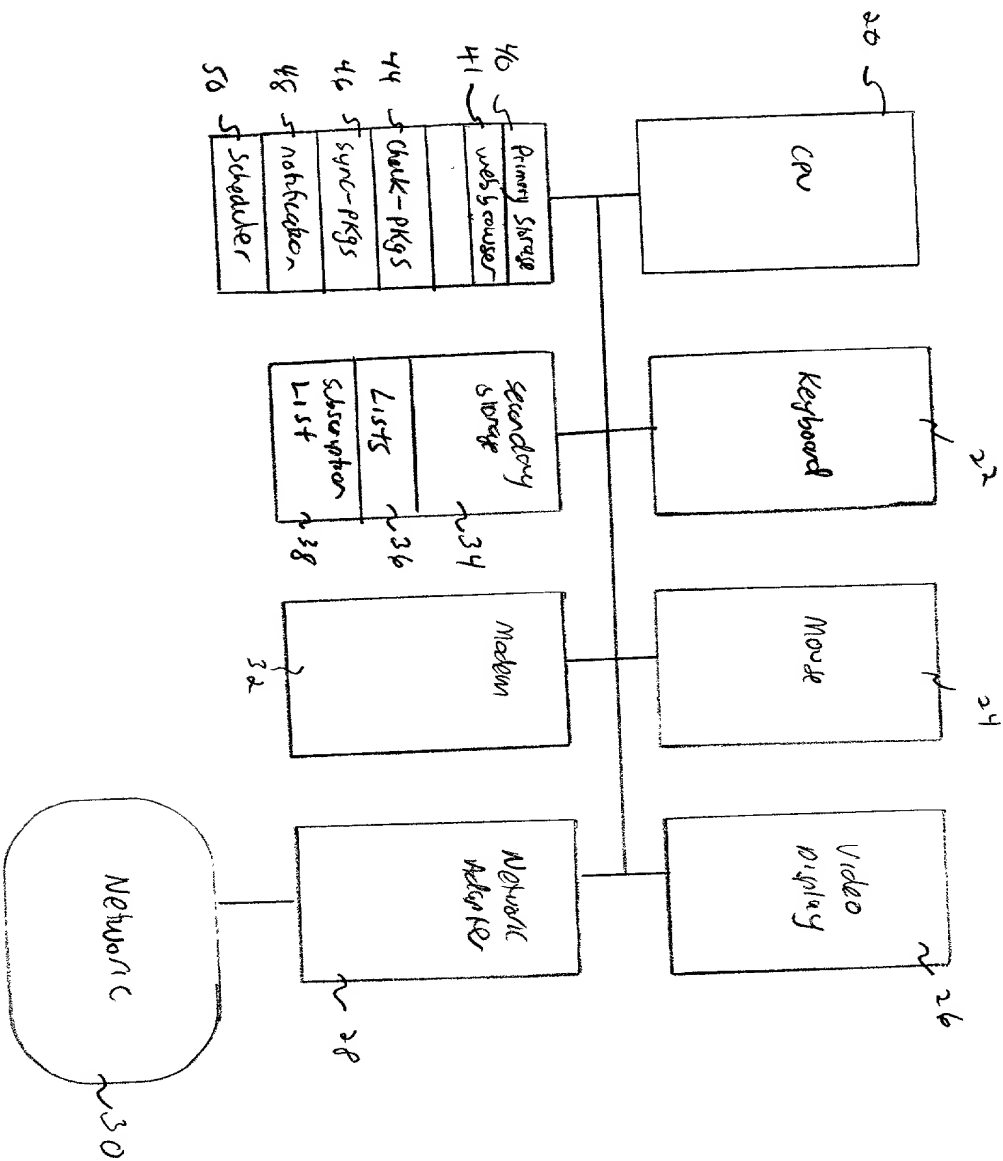


Figure 1



← 14

Figure 2

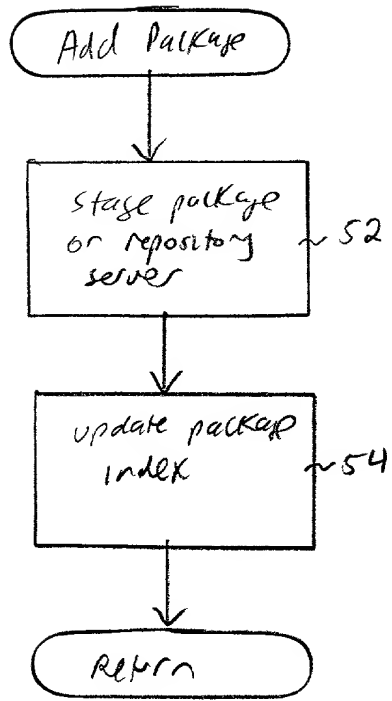


Figure 3

Name	Version	Type	Location
coyote	2.0	D	http://www.arizona.com/coyote
roadrunner	1.2	R	http://www.arizona.com/roadrunner
rocketstates	1.0	P - wecoyote	ftp://ftp.ace.com/sets/rocketstates
dynamic	1.1	D	location 1, location 2, location 3

Package Index 60

Figure 4

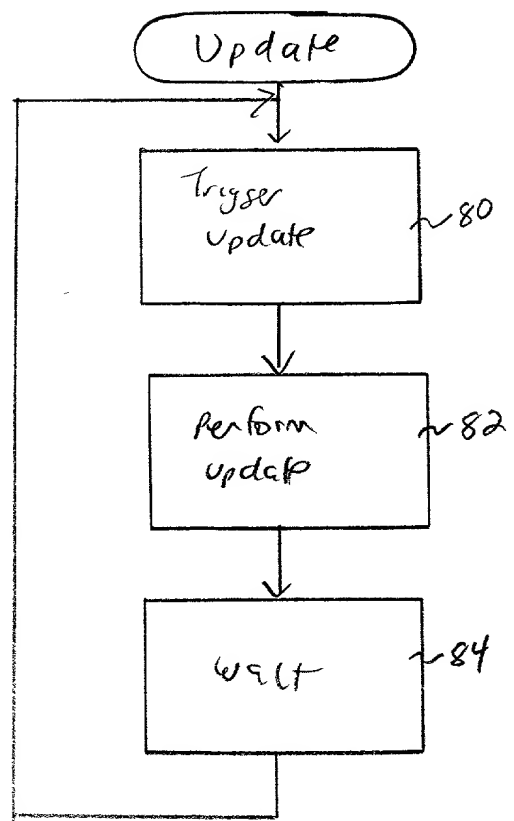


figure 5

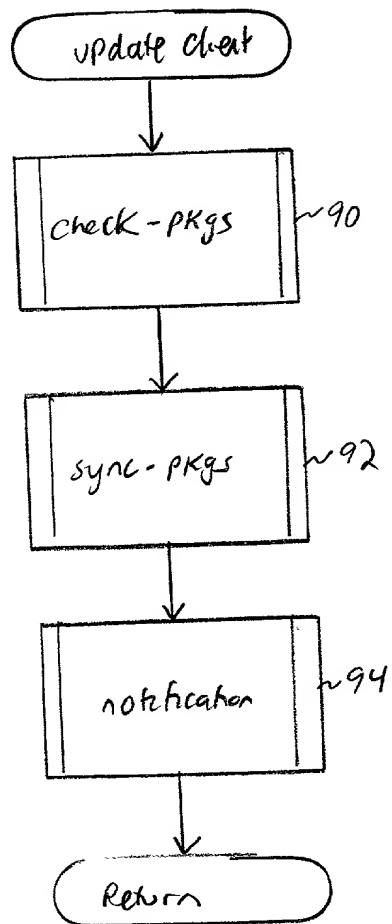


Figure 6

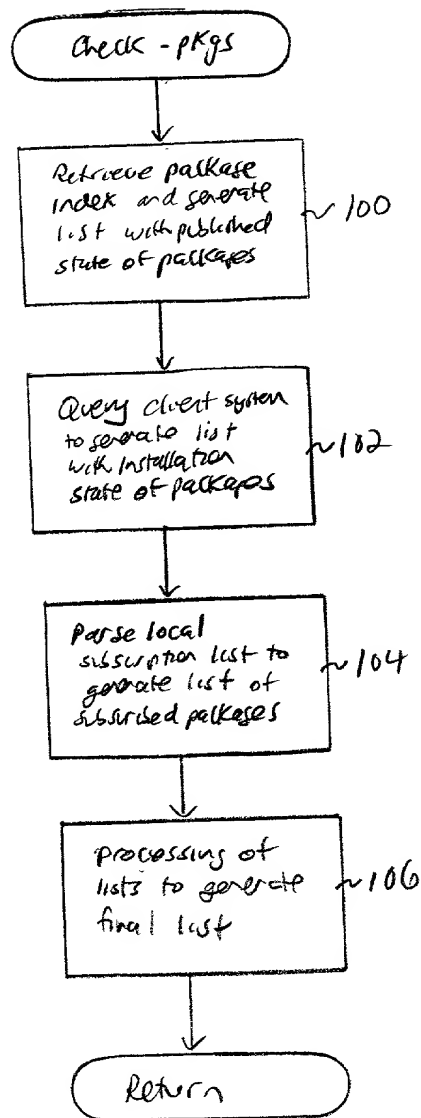


Figure 7

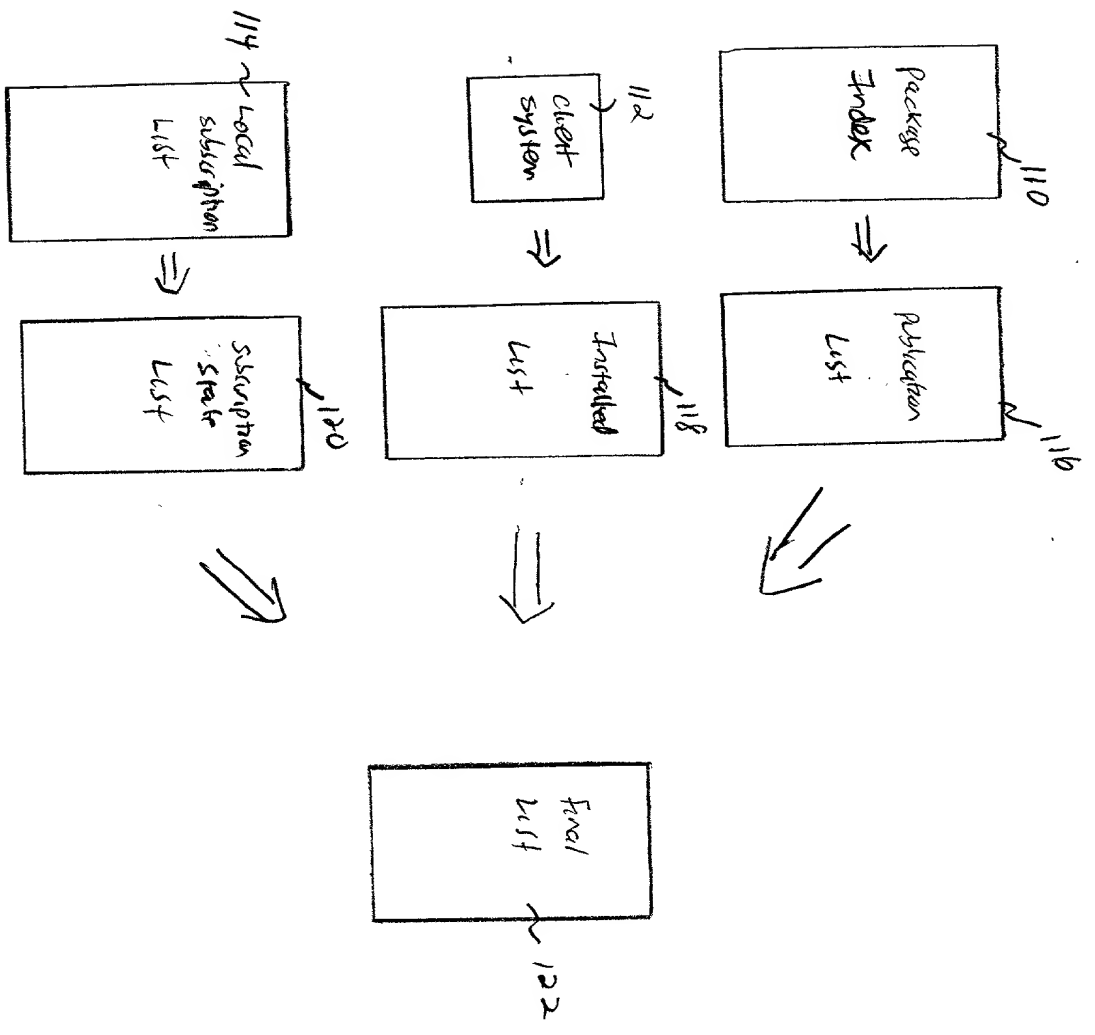


Figure 8

Subscription List, 114

package1
package 4
package5

Figure 9

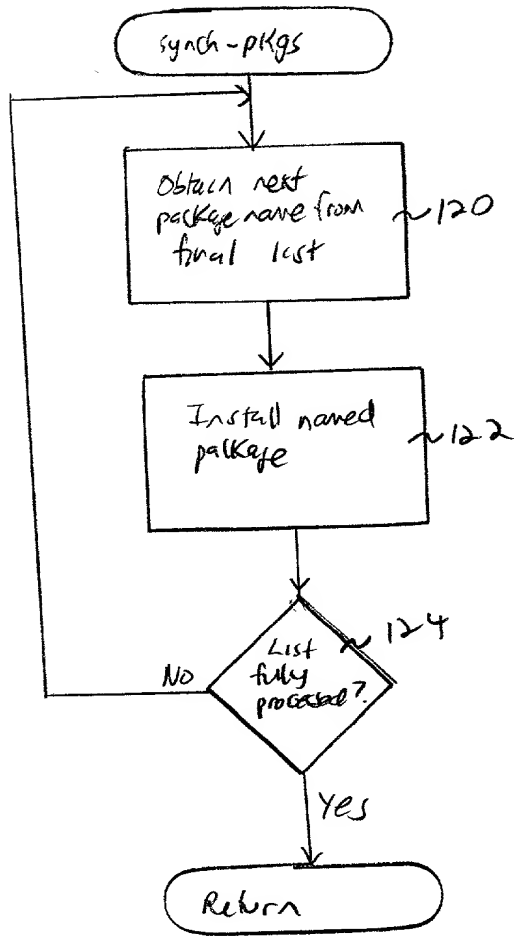


Figure 10

[illegible]

Package	Repository Server	Errors
packagel	Repository 2	None
packages	Repository 1	Insufficient memory

Notafzcytzon 130

Figure 11

600 400 300 200 100

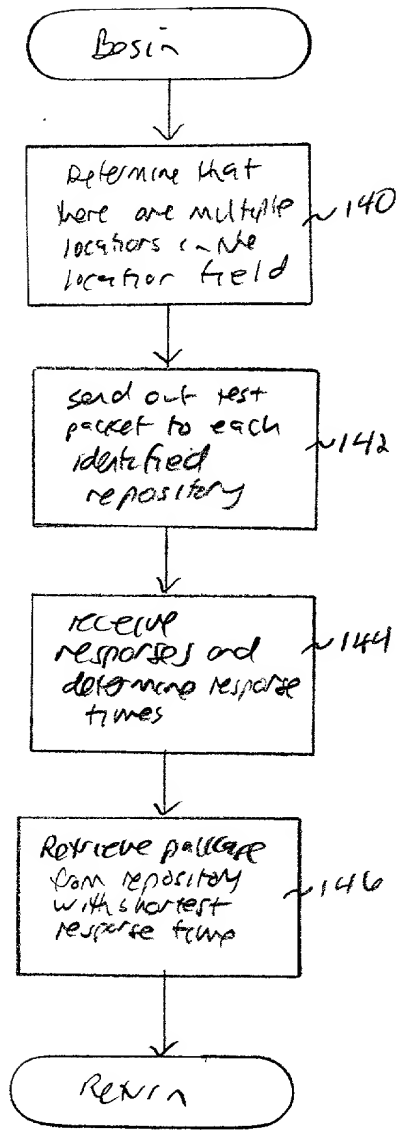


Figure 12

[illegible]

Attorney's
Docket
Number SMQ-011
(P3610/RSH)

Declaration, Petition and Power of Attorney for Patent Application

As a below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below next to my name,

I believe I am the original, first and sole inventor (if only one name is listed below) or an original, first and joint inventor (if plural names are listed below) of the subject matter which is claimed and for which a patent is sought on the invention entitled

LIGHTWEIGHT GLOBAL DISTRIBUTION MECHANISM
the specification of which

(check one)

X is attached hereto.

was filed on _____ as _____

Application Serial No.

and was amended on _____
(if applicable)

I do not know and do not believe that the subject matter of this application was known or used by others in the United States or patented or described in a printed publication in any country before my invention thereof, or patented or described in a printed publication in any country or in public use or on sale in the United States more than one year prior to the date of this application, or first patented or caused to be patented or made the subject of an inventor's certificate by me or my legal representatives or assigns in a country foreign to the United States prior to the date of this application on an application filed more than twelve months (six months if this application is for a design) before the filing of this application; and I acknowledge my duty to disclose information of which I am aware which is material to the examination of this application, that no application for patent or inventor's certificate on the subject matter of this application has been filed by me or my representatives or assigns in any country foreign to the United States, except those identified below, and that I have reviewed and understand the contents of the specification, including the claims as amended by any amendment referred to herein.

I acknowledge the duty to disclose to the Office all information known to me to be material to patentability as defined in Title 37, Code of Federal Regulations, §1.56.

CLAIM OF BENEFIT OF EARLIER FOREIGN APPLICATION(S)

I hereby claim priority benefits under Title 35, United States Code, §119 of any foreign application(s) for patent or inventor's certificate listed below, and have also identified below any foreign application(s) for patent or inventor's certificate filed by me on the same subject matter having a filing date before that of the application(s) from which priority is claimed.

Check one:

☒ no such applications have been filed.

☐ such applications have been filed as follows

EARLIEST FOREIGN APPLICATION(S), IF ANY, FILED WITHIN 12 MONTHS
(6 MONTHS FOR DESIGN) PRIOR TO THIS U.S. APPLICATION

Country	Application Number	Date of Filing (month,day,year)	Priority Claimed Under 35 USC 119
			<input type="checkbox"/> Yes <input type="checkbox"/> No
			<input type="checkbox"/> Yes <input type="checkbox"/> No
			<input type="checkbox"/> Yes <input type="checkbox"/> No
			<input type="checkbox"/> Yes <input type="checkbox"/> No
			<input type="checkbox"/> Yes <input type="checkbox"/> No

ALL FOREIGN APPLICATION(S), IF ANY FILED MORE THAN 12 MONTHS
(6 MONTHS FOR DESIGN) PRIOR TO THIS U.S. APPLICATION

CLAIM FOR BENEFIT OF U.S. PROVISIONAL APPLICATION(S)

I hereby claim the benefit under 35 U.S.C. §119(e) of any United States provisional application(s) listed below.

(Application Serial No.)

(Filing Date)

(Application Serial No.)

(Filing Date)

CLAIM FOR BENEFIT OF EARLIER U.S./PCT APPLICATION(S)

I hereby claim the benefit under Title 35, United States Code, §120 of any earlier United States application(s) or PCT international application(s) designating the United States listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the earlier application(s) in the manner provided by the first paragraph of Title 35, United States Code, §112, I acknowledge the duty to disclose to the Office all information known to me to be material to patentability as defined in Title 37, Code of Federal Regulations, §1.56 which became available between the filing date(s) of the earlier application(s) and the national or PCT international filing date of this application. As to subject matter of this application which is common to my earlier application(s), if any, described below, I do not know and do not believe that the same was known or used by others in the United States or patented or described in a printed publication in any country before my invention thereof, or patented or described in a printed publication in any country or in public use or on sale in the United States more than one year prior to the date(s) of said earlier application(s), or first patented or caused to be patented or made the subject of an inventor's certificate by me or my legal representatives or assigns in a country foreign to the United States prior to the date(s) of said earlier application(s) on an application filed more than twelve months (six months if this application is for a design) before the filing of said earlier application(s); and I acknowledge that no application for patent or inventor's certificate on said subject matter has been filed by me or my representatives or assigns in any country foreign to the United States except those identified herein.

<u>(Application Serial No.)</u>	<u>(Filing Date)</u>	<u>(Status)</u> (patented,pending,aband.)
<u>(Application Serial No.)</u>	<u>(Filing Date)</u>	<u>(Status)</u> (patented,pending,aband.)

POWER OF ATTORNEY: As a named inventor, I hereby appoint the following attorneys and/or agents to prosecute this application and transact all business in the Patent and Trademark Office connected therewith.

W. Hugo Liepmann	Reg. No. 20,407	Lawrence E. Monks	Reg. No. 34,224
James E. Cockfield	Reg. No. 19,162	David A. Lane, Jr.	Reg. No. 39,261
Thomas V. Smurzynski	Reg. No. 24,798	Catherine J. Kara	Reg. No. 41,106
Ralph A. Loren	Reg. No. 29,325	Linda M. Chinn	Reg. No. 31,240
Giulio A. DeConti, Jr.	Reg. No. 31,503	Faustino A. Lichauco	Reg. No. 41,942
Ann Lamport Hammitte	Reg. No. 34,858	Jeanne M. DiGiorgio	Reg. No. 41,710
Elizabeth A. Hanley	Reg. No. 33,505	Megan E. Williams	Reg. No. 43,270
Amy E. Mandragouras	Reg. No. 36,207	Nicholas P. Triano III	Reg. No. 36,397
John V. Bianco	Reg. No. 36,748	Peter C. Lauro	Reg. No. 32,360
Anthony A. Laurentano	Reg. No. 38,220	Reza Mollaaghababa	Reg. No. P43,810
Jane E. Remillard	Reg. No. 38,872	Timothy J. Douros	Reg. No. 41,716
Jeremiah Lynch	Reg. No. 17,425	John L. Welch	Reg. No. 28,129
Kevin J. Canning	Reg. No. 35,470	Philip J. McKay	Reg. No. 38,966
Kenneth Olsen	Reg. No. 26,493	Robert S. Hauser	Reg. No. 37,847
Matthew C. Rainey	Reg. No. 32,291	Joseph T. FitzGerald	Reg. No. 33,881
Timothy J. Crean	Reg. No. 37,116	Alexander E. Silverman	Reg. No. 37,940

Send Correspondence to Kevin J. Canning, Esq. at **Customer Number: 000959** whose address is:

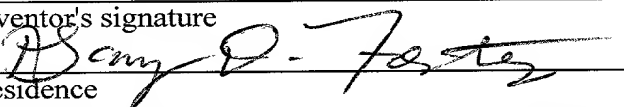
Lahive & Cockfield, LLP, 28 State Street, Boston, MA 02109

Direct Telephone Calls to: (name and telephone number)

Kevin J. Canning, Esq., (617) 227-7400

Wherefore I petition that letters patent be granted to me for the invention or discovery described and claimed in the attached specification and claims, and hereby subscribe my name to said specification and claims and to the foregoing declaration, power of attorney, and this petition.

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

Full name of sole or first inventor Gary D. Foster	
Inventor's signature 	Date March 12, 1999
Residence 962 Kiely Boulevard, Unit H, Santa Clara, California, 95051	
Citizenship United States	
Post Office Address (if different) Same as Above	